 .

Realizado por:

John Clever Choque Aranibar

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tutor: (MSC. Ing., Marcelo Ticona Barroso)

###### 

###### Relator: Miguel Angel Ávila

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

2024

Universidad Católica Boliviana “San Pablo”

Facultad de Ingeniería

Departamento de Ingeniería de Sistemas

##### La Paz - Bolivia



### **“CONTROL Y SEGUIMIENTO DE INVENTARIOS PARA EQUIPOS ELECTRÓNICOS MEDIANTE UNA APLICACIÓN MÓVIL”**

**CASO UNIVERSIDAD CATÓLICA BOLIVIANA “SAN PABLO”**

PROYECTO DE GRADO PRESENTADO AL DEPARTAMENTO DE CENTRO DE COMPUTO PARA EL MONITOREO Y SEGUIMIENTO DE LOS SITEMAS DE INFORMACIÓN

TRA

1. **MARCO PREFERENCIAL**
   1. **INTRODUCCIÓN**

El presente proyecto de grado propone realizar un control y seguimiento de inventarios para los equipos electrónicos. Para la Universidad Católica Boliviana “San Pablo”, el control de inventarios de equipos electrónicos es una tarea muy importante para asegurar el correcto funcionamiento de las actividades académicas

Por otro lado, se trata de una función que resulta particularmente desafiante de optimizar, también existen varios sistemas de inventarios como ser (Shopify, inventario de plazas y Shopkeep), sin embargo, estos se limitan a listar y al mantenimiento de los equipos tecnológicos.

El departamento de Centro de Cómputo y CSinf de la Universidad Católica Boliviana busca optimizar el proceso de inventarios de los equipos tecnológicos.

Por tal motivo, el presente proyecto de grado consiste en desarrollar un prototipo de Sistema de control seguimiento y mantenimiento de equipos electrónicos en aplicación móvil

* 1. **ANTECEDENTES**
     1. **ANTECEDENTES DEL OBJETO DE ESTUDIO**
     2. **REFERENCIAS TÉCNICAS DE OTROS TRABAJOS**

En el contexto internacional son más los casos de empresas que desarrollaron plataformas para el control de soporte de equipos electrónicos a continuación se presentaran casos que hasta la redacción de este documento están disponibles

El Departamento de Ingeniería del Hospital Central de la Fuerza Aérea del Perú se seleccionó como objeto de estudio, dado que presenta varias deficiencias administrativas en sus procesos internos, incluyendo la recepción, registro y cierre de Órdenes de Trabajo, así como el mantenimiento preventivo y correctivo de los equipos hospitalarios del HCFAP.

Además, se toma en cuenta el proyecto "Desarrollo de Aplicación Móvil para el Registro y Seguimiento del Mantenimiento Correctivo de los Equipos Biomédicos", el cual expone el diseño de una aplicación por medio de una plataforma online llamada App Inventor, para el control y seguimiento del mantenimiento correctivo en él Hospital Departamental San Vicente de Paúl. Este proyecto utiliza diversas estrategias e investigaciones para mejorar el registro, control y seguimiento del mantenimiento correctivo de los equipos biomédicos. La aplicación diseñada facilita la consulta y reporte de daños en equipos biomédicos, contribuyendo así a la mejora de los procesos de ingeniería biomédica en la institución. (Mendoza, 2017)

La universidad politécnica de Madrid.

Logimov es un sistema de inventarios y seguimientos tanto en equipos tecnológicos como en productos así mismo Logimov tiene su sistema realizado en Web y aplicación móvil es eficiente a la hora de realizar cualquier tipo de mantenimiento o seguimiento de productos o equipos tecnológicos. (Flórez, n.d.)

En un artículo de Revista Ingeniería e Innovación del Futuro (RIIF) propone el desarrollo de una aplicación móvil personalizada diseñada para la gestión eficiente de inventarios de productos en equipos tecnológicos de la empresa SOLINTEG360. La plataforma móvil fue construida utilizando MAUI, un trabajo multiplataforma que es compatible con múltiples sistemas operativos garantizando una experiencia uniforme para los usuarios en dispositivos iOS y Android.

SOLITEG360 permite realizar un seguimiento en tiempo real de sus existencias, y mejorar la eficiencia operativa ofreciendo a la empresa una herramienta personalizada y adaptable a sus necesidades específicas de gestión de inventarios.

(Escobar, 2023)

* 1. **DESCRIPCIÓN DEL OBJETO DE ESTUDIO**
     1. **CENTRO DE COMPUTO DE LA UNIVERSIDAD CATÓLICA BOLIVIANA “SAN PABLO”**

En la Universidad Católica Boliviana "San Pablo", el departamento de Centro de Cómputo juega un papel importante. Garantizar el óptimo funcionamiento y rendimiento de los equipos tecnológicos de la institución es su principal responsabilidad. El departamento se encarga de realizar tareas esenciales como el mantenimiento preventivo y correctivo, el seguimiento y el control, así como la protección de todos los dispositivos informáticos.

* + 1. **EL PROCESO DE MANTENIMIENTO DE LOS EQUIPOS DE LA UNIVERSIDAD CATÓLICA BOLIVIANA “SAN PABLO”**

Se debe seguir un proceso estructurado para realizar el mantenimiento de los equipos. El primer paso en este proceso es encontrar el equipo que necesita mantenimiento. Posteriormente, el problema o falla que causa el mal funcionamiento del dispositivo se identifica. El mantenimiento se lleva a cabo inmediatamente en el mismo lugar si se trata de una falla de menor complejidad. Sin embargo, se requiere un mantenimiento más exhaustivo si el equipo presenta fallas más complicadas. En estos casos, el encargado de soporte del centro de computación debe obtener el permiso correspondiente para llevar el equipo con falla al laboratorio del departamento. En ese lugar, el personal capacitado y experimentado realizará el mantenimiento necesario para que vuelva a funcionar correctamente. Una vez completado el proceso, el equipo se devolverá a su ubicación inicial y se proporcionará una breve explicación de la falla encontrada y las medidas tomadas para solucionarla.

* + 1. **CONTROL DE INVENTARIO DE LOS EQUIPOS DE LA UNIVERSIDAD CATÓLICA BOLIVIANA “SAN PABLO”**

En la Universidad Católica Boliviana "San Pablo", se realiza un seguimiento del inventario de los equipos tecnológicos. Este procedimiento requiere el registro completo de todos los dispositivos, organizándolos por departamento y bloque. Además, se realiza un inventario de los componentes internos y externos de cada equipo, lo que permite obtener una imagen completa de su configuración y estado actual.   
Este método de registro y seguimiento de equipos tecnológicos tiene muchas ventajas. Para empezar, facilita la identificación precisa de cada dispositivo y sus características, lo que optimiza la gestión y asignación de recursos. Además, tener información detallada sobre los componentes permite un mantenimiento preventivo más efectivo y anticipa posibles fallas o necesidades de actualización.

El control de inventario también permite mantener un seguimiento continuo del estado de cada equipo e identificar aquellos que requieren atención prioritaria o reemplazo. Este proceso ayuda a maximizar la vida útil de los dispositivos y garantizar un rendimiento óptimo en todo momento, lo que permite que las operaciones académicas y administrativas se mantengan en curso.

* 1. **IDENTIFICACIÓN DEL PROBLEMA**
     1. **PLANTEAMIENTO DEL PROBLEMA**

El Centro de Cómputo de la Universidad Católica Boliviana "San Pablo" trabaja manualmente en el inventario, mantenimiento y seguimiento de los sistemas tecnológicos, lo que plantea algunos problemas y restricciones para la institución.   
En cuanto al mantenimiento de equipos, el proceso implica que cualquier personal administrativo solicite al encargado del Centro de Cómputo que confirme la necesidad de mantenimiento antes de realizar un mantenimiento previo. Después, el equipo de soporte lleva el dispositivo al laboratorio para completar las tareas de mantenimiento. Sin embargo, este procedimiento causa demoras.

Cuando se descubre una falla en uno de los bloques de la universidad, surge otro problema recurrente. Durante estas circunstancias, Csinf designa a los empleados del Centro de Computación para llevar a cabo una revisión inicial de la falla. Debido a la falta de experiencia, estos empleados suelen demorar en la identificación y resolución del problema, lo que resulta en molestias y pérdida de tiempo para los docentes o personal afectado, quienes no pueden continuar con sus actividades habituales.

Esto suele ocurrir cuando llegan nuevos miembros al departamento del Centro de Cómputo.   
Además, el Centro de Cómputo enfrenta un gran desafío al manejar manualmente el inventario de los equipos tecnológicos. Todos los dispositivos deben ser registrados y documentados por el personal designado. Errores, retrasos y falta de información son posibles durante este proceso manual. Estás restricciones reducen el rendimiento y la eficiencia de las actividades del personal administrativo de la universidad.

Los retrasos en el mantenimiento y la falta de información precisa sobre el estado del equipo pueden interrumpir el flujo de trabajo.

Debido a los procesos manuales actuales, el Centro de Cómputo de la Universidad Católica Boliviana "San Pablo" enfrenta dificultades importantes en el control de sus recursos tecnológicos. La implementación de una aplicación móvil automatizada para modernizar estos procesos podría mejorar significativamente la eficiencia, precisión y agilidad en el mantenimiento, seguimiento e inventario de los equipos, lo que optimizaría el rendimiento general de las operaciones de la institución.

**DIAGRAMA DE ISHIKAWA**

Diagrama

Descripción generada automáticamente

***Figura 1:*** *Diagrama de Ishikawa*

* 1. **FORMULACIÓN DEL PROBLEMA**

¿Cómo podría optimizarse el proceso de control y seguimiento de inventarios para equipos electrónicos en la Universidad Católica Boliviana “San Pablo” mediante una aplicación móvil?

* 1. **OBJETIVOS**
     1. **OBJETIVO GENERAL**

Desarrollar una aplicación móvil para optimizar los procesos de control y seguimiento de inventarios en equipos electrónicos en la “Universidad Católica Boliviana”.

* + 1. **OBJETIVOS ESPECÍFICOS**
* Diseñar una interfaz en Flutter intuitiva para el registro y actualización de los equipos electrónicos
* Desarrollar un módulo para el seguimiento de equipos electrónicos mediante código de barra
* Generar reporte detallado sobre el inventario de equipos, su estado. Y seguimiento y control de los equipos
* Diseñar e implementar una base de datos que soporte los requerimientos del control y seguimiento del proyecto.
  1. **JUSTIFICACIONES**
     1. **JUSTIFICACIÓN PRÁCTICA**

Para asegurar un entorno académico ideal en la Universidad Católica Boliviana "San Pablo", es necesario administrar de manera efectiva los recursos tecnológicos. Las herramientas informáticas son esenciales para el aprendizaje, la investigación y el desarrollo de actividades educativas en todos los niveles y áreas de estudio. El mantenimiento adecuado, el seguimiento preciso y el control de inventario actualizado aseguran que estos recursos estén disponibles y funcionen correctamente, evitando interrupciones que puedan afectar el proceso de enseñanza-aprendizaje. Además, el uso de sistemas automatizados mejoraría el uso de los recursos tecnológicos en la educación y fomentaría la innovación y el uso de tecnologías avanzadas.

* + 1. **JUSTIFICACIÓN SOCIAL**

La Universidad Católica Boliviana "San Pablo" tiene un impacto en la sociedad, formando profesionales y generando conocimiento. Una gestión deficiente de los recursos tecnológicos puede repercutir negativamente en la calidad de la educación impartida y, por lo tanto, en la preparación de los futuros profesionales. Contar con sistemas automatizados para el mantenimiento, seguimiento e inventario de los equipos informáticos contribuiría a mejorar la eficiencia de los servicios educativos, asegurando que los estudiantes reciban una formación de calidad y adquieran las habilidades necesarias para enfrentar los desafíos del mundo laboral y social actual.

* + 1. **JUSTIFICACIÓN ECONÓMICA**

Los equipos informáticos representan una inversión considerable para la Universidad Católica Boliviana "San Pablo". Un mantenimiento adecuado y un seguimiento preciso pueden prolongar la vida útil de estos recursos, evitando costos adicionales por reparaciones o reemplazos prematuros. Además, la automatización de los procesos de mantenimiento, seguimiento e inventario puede reducir los costos Asociados a la gestión manual, optimizando el uso de los recursos humanos y financieros de la institución. Al mismo tiempo, una correcta gestión de los recursos tecnológicos puede mejorar la productividad del personal administrativo y académico,

* 1. **LÍMITES Y ALCANCES**
     1. **LÍMITES**
* El sistema se enfocará exclusivamente en el seguimiento y mantenimiento de equipos electrónicos de la Universidad Católica Boliviana
* No contemplará la gestión de otro tipo de activos o inventarios
* En esta fase, no incluirá integración con sistemas externos o proveedores
* No se realizará el prototipo en un entorno Web
  + 1. **ALCANCES**
* Aplicación móvil para el personal encargado del monitoreo y control de equipos
* Registro, actualización, búsqueda, seguimiento y mantenimiento de equipos por categorías como ser. Código, departamento, bloque y responsable
* Operación de asignación de equipos a áreas o dependencias específicas
* Seguimiento de movimientos, préstamos y traslados de equipos entre ubicaciones
* Registro de mantenimientos preventivos y correctivos realizados
* Sistema de Login con Google
  1. **METODOLOGÍAS DE INVESTIGACIÓN**
     1. **MÉTODOS**

**UP (Process undefind)**

Para la ejecución del proyecto, se ha optado por emplear la metodología de proceso unificado. Esta elección en la necesidad de realizar un análisis exhaustivo de los requerimientos del proyecto. Tal análisis permitirá establecer una arquitectura sólida y adecuada para el desarrollo, Asimismo la metodología de proceso unificado asegurará que se genere una documentación detallada y precisa que refleje de manera completa los aspectos y procesos involucrados en el proyecto

**Bloc**

La arquitectura Bloc se destaca por su capacidad para separar claramente las responsabilidades y la lógica de una aplicación. Esto simplifica la reutilización de código, facilita la modularidad y promueve la escalabilidad del proyecto. Al usar streams de datos para la comunicación entre componentes, Bloc permite una gestión ágil de eventos y cambios de estado. En resumen, Bloc es una herramienta sólida para el desarrollo de aplicaciones modernas que requieren alta modularidad y rendimiento.

* + 1. **MODELOS**

**UML(LanguageModelUndefind)**  
El UML (Lenguaje Unificado de Modelado) es un estándar utilizado en ingeniería de software para visualizar, especificar y documentar sistemas de software. Proporciona una variedad de diagramas gráficos que representan aspectos clave de un sistema, como su estructura estática y su comportamiento dinámico. Estos diagramas ayudan a los desarrolladores a comprender, comunicar y colaborar en el diseño y desarrollo de software de manera efectiva.

Para el diseño y análisis del proyecto se hará los diagramas de Casos de Uso, Actividades, Clase y de Secuencia. (Education, 2017.)

* + 1. **HERRAMIENTAS**

|  |  |
| --- | --- |
| **PROGRAMA** | **DESCRIPCIÓN** |
| Bizagi | “Es una herramienta que permite gestionar el ciclo de vida completo del modelamiento de procesos del negocio de una manera fácil y amigable.” (bizagi 2021) |
| Whimsical | Whimsical es el espacio de trabajo visual que ayuda a tu equipo a comunicarse a la velocidad del pensamiento ya que ofrece esquemas de base, diagramas de flujo, mapas mentales, notas adhesivas y documentos (beta) que son increíblemente rápidos. (asana, 2022) |
|  | “Enterprise Architect es una herramienta comprensible de diseño y análisis UML, cubriendo el desarrollo de software desde el paso de los requerimientos a través de las etapas del análisis, modelos de diseño, pruebas y mantenimiento. EA es una herramienta multi-usuario, basada en Windows, diseñada para ayudar a construir software robusto y fácil de mantener. Ofrece salida de documentación flexible y de alta calidad.” (Ecured 2021) |
|  | “Office 2019 es la suite ofimática mínima y básica de Microsoft, el Office de toda la vida que no incluye almacenamiento en la nube ni sincronizaciones ni muchas de las funciones que hacen especial a Office 365. Es simplemente una suite con Word, Excel y PowerPoint, que pagas una vez y ya tienes la licencia para utilizarlo siempre.” (xataka 2021) |
| Vertabelo | “Herramienta en línea para el diseño visual de bases de datos. Esta aplicación web admite siete motores de bases de datos principales: PostgreSQL, MySQL, Microsoft SQL Server, SQLite, Oracle 11g, IBM DB2 y HSQLDB.” (progsoft 2021) |

**Tabla 1: Herramientas de modelado**

Las herramientas utilizadas para el desarrollo del proyecto son:

|  |  |
| --- | --- |
| **PROGRAMAS** | **DESCRIPCIÓN** |
| IntelliJ IDEA | “IntelliJ IDEA es un IDE inteligente y sensible al contexto para trabajar con Java y otros lenguajes JVM como Kotlin, Scala y Groovy en todo tipo de aplicaciones.” (Jetbrains 2021) |
|  | “Spring es un marco de trabajo o Framework que nos permite desarrollar aplicaciones con el Lenguaje de Programación Java. Spring Framework cuenta con muchas características que hacen a los desarrolladores más productivos.” (nube colectiva 2021) |
| PostgreSQL | “PostgreSQL, o simplemente Postgres para darle un nombre más pintoresco, es un sistema de código abierto de administración de bases de datos del tipo relacional, aunque también es posible ejecutar consultas que sean no relaciones. En este sistema, las consultas relacionales se basan en SQL, mientras que las no relacionales hacen uso de JSON.” (infranetworking2021) |
| Flutter | “Flutter es el kit de herramientas de UI de Google para realizar hermosas aplicaciones, compiladas nativamente, para móvil, web y escritorio desde una única base de código.” (esflutter 2021) |
| Dart | “Dart es un lenguaje de programación open source, relativamente nuevo, que fue desarrollado por Google y que lanzó su primera versión en 2011. Este lenguaje se creó con el objetivo de permitir a los desarrolladores utilizar un lenguaje orientado a objetos y con análisis estático de tipo.” (hiberus 2021) |
| Visual Studio Code | “Visual Studio Code es un editor de código fuente desarrollado por Microsoft para Windows, Linux y macOS. Incluye soporte para la depuración, control integrado de Git, resaltado de sintaxis, finalización inteligente de código, fragmentos y refactorización de código.” (Wikipedia 2021) |
| Android Studio | Android Studio es el entorno de desarrollo integrado (IDE) oficial para el desarrollo de aplicaciones Android, desarrollado por Google. Proporciona herramientas potentes para el diseño de interfaces de usuario, la depuración y la compilación de aplicaciones Android. |
| Windowns | Windows es un sistema operativo desarrollado por Microsoft, ampliamente utilizado en computadoras personales y servidores, conocido por su interfaz gráfica de usuario intuitiva, su compatibilidad con una amplia gama de hardware y su amplia disponibilidad de software. |
| Rest API | Una API, o interfaz de programación de aplicaciones, se adapta a los principios de la arquitectura REST, lo que facilita la comunicación entre diferentes servicios web. Funciona como un contrato entre el cliente y el proveedor, estableciendo los datos necesarios para el cliente y los requerimientos del proveedor. REST, por otro lado, es un estilo de arquitectura de software que permite la conexión de sistemas mediante el protocolo HTTP. Esto posibilita la obtención y generación de datos y operaciones en formatos como JSON o XML, siendo JSON el más utilizado debido a su legibilidad y peso ligero. (Rosa, 218) |
| ORM | ORM significa Mapeo Objeto-Relacional (Object-Relational Mapping, en inglés), es una técnica de programación que permite convertir datos entre sistemas de tipos de datos incompatibles en bases de datos relacionales y lenguajes de programación orientados a objetos.  ORM permite trabajar con la base de datos utilizando objetos en lugar de sentencias SQL directas. Esto simplifica el proceso de interacción con la base de datos y reduce la cantidad de código necesario para realizar operaciones CRUD (Crear, Leer, Actualizar, Eliminar). Además, ORM ayuda a mantener la consistencia de los datos y a reducir la posibilidad de errores en la lógica de acceso a la base de datos. (Díaz, 2021) |
| **Scrumban** | Scrumban es una metodología de desarrollo ágil surge de la combinación de las técnicas de desarrollo Kanban y Scrum. Se basa en la flexibilidad de Kanban para permitir que los equipos de trabajo mejoren continuamente mientras combina las características de Scrum para ser ágil. Por tener actividades dinámicas, resulta muy útil para el desarrollo de software y para ambientes donde se presentan cambios constantes. (Atlassian, s.f.) |

* 1. **ANÁLISIS PRELIMINAR DEL PROYECTO**

El departamento de Centro de Cómputo de la Universidad Católica Boliviana "San Pablo" ha tenido dificultades para manejar sus recursos tecnológicos. Hasta ahora, todas las tareas relacionadas con el mantenimiento, el seguimiento e inventario de computadoras se realizan manualmente. Esto ha provocado problemas que afectan negativamente la eficiencia y la productividad, incluidos:

* **Procesos manuales lentos y propensos a errores**

El mantenimiento, el seguimiento e inventario de los equipos informáticos se realizan manualmente, lo que aumenta la carga de trabajo, los retrasos y los riesgos de errores en el manejo de la información.

* **Falta de control y seguimiento en tiempo real**

No hay un monitoreo continuo y actualizado del estado de los dispositivos, lo que dificulta la detección temprana de fallas o problemas y retrasa las acciones correctivas necesarias.

* **Falta de automatización y tecnologías avanzadas**

Los procesos de mantenimiento, seguimiento e inventario no se están automatizando y optimizando, utilizando tecnologías disponibles, como aplicaciones móviles.

* 1. **PROPUESTA DE SOLUCIÓN**

La solución consiste en crear una aplicación para dispositivos móviles que facilite el control de los recursos tecnológicos. Esta aplicación móvil facilitará el registro equipos electrónicos, permitiendo un seguimiento detallado de cada dispositivo y asignar nuevos equipos al personal administrativo.

El usuario podrá registrar sus equipos y monitorear y asignar nuevos equipos al personal administrativo. Esto registrará los equipos asignados tanto internos como externos para obtener más información y tener los repuestos disponibles para realizar los mantenimientos correspondientes

Diagrama

Descripción generada automáticamente

Figura 2: Diagrama tecnológico

El diagrama tecnológico del proyecto actual se muestra en la figura 2. Este proyecto utilizará una arquitectura cliente-servidor con un front-end y un back-end. Un sistema móvil desarrollado en Flutter será la parte front-end, que enviará peticiones JSON al servidor que alojará una base de datos SQL PostgreSQL. El back-end, desarrollado en Spring boot, responderá a las peticiones JSON enviadas desde el front-end.

* 1. **CRONOGRAMA**

1. **MARCO TEÓRICO**
   1. **UNFIED PROCESS (UP)**
   2. **METODOLOGÍAS DE ANÁLISIS Y DISEÑO**
      1. **MODELO CASCADA**

El modelo de cascada de desarrollo es un procedimiento lineal que divide los procesos de desarrollo en fases consecutivas del proyecto. En contraste con los modelos iterativos, cada una de estas fases se lleva a cabo una sola vez. Ellos

Los resultados de cada paso sirven como base para la siguiente. El modelo de cascada se utiliza principalmente en el desarrollo de software. Ionios, 2021

Este modelo se atribuye al teórico de la información Winston W. Royse. Royse crea un modelo iterativo incremental en el que cada una de las fases se basa en la fase anterior y verifica los resultados de esta base.

Interfaz de usuario gráfica, Diagrama

Descripción generada automáticamente

Figura 3: Modelo Cascada

El ciclo de vida del modelo de cascada se muestra en la ilustración 3. Se compone de cinco pasos: análisis, diseño, implementación, verificación y mantenimiento. Cada paso del modelo se explica a continuación.

* + 1. **ANÁLISIS**

Como todo proyecto de software comienza con una fase de análisis que incluye una definición de requisitos y un estudio de viabilidad. En este estudio de viabilidad se avalúan los costes, factibilidad y rentabilidad del proyecto de software. Después, se realiza la definición detallada de los requisitos, el cual da como resultado un pliego de condiciones y una descripción detallada de cómo se han de cumplir los requisitos. Por último, también incluye un análisis de la definición de los requisitos en el que los problemas complejos se dividen en pequeñas tareas secundarias. (ionos, 2021)

* + 1. **DISEÑO**

En esta fase, se desarrolla una solución específica que cumple con los requisitos establecidos en la etapa anterior. En este lugar se elabora la arquitectura del software y se elabora un plan de diseño detallado en función de elementos específicos como interfaces, entornos de trabajo y bibliotecas. (ionos, 2021)

* + 1. **IMPLEMENTACIÓN**

En esta fase se ejecuta la arquitectura de software planteado en la fase de diseño, en la que incluye la búsqueda de errores, pruebas unitarias y la programación del software. Este mismo da como resultado un producto de software que se comprueba por primera vez como producto final en la siguiente fase. (ionos, 2021)

* + 1. **VERIFICACIÓN**

En esta fase se incluye la integración del software en el entorno seleccionado, por norma general los productos de software se envían a usuario finales en versión beta. Las pruebas de aceptación desarrolladas en la primera fase permiten verificar si el software cumple correctamente con las exigencias planteadas anteriormente. (ionos, 2021)

* + 1. **MANTENIMIENTO**

En esta última fase se autoriza la aplicación productiva del software, una vez que la fase de verificación concluya con éxito. Esta fase del modelo en cascada incluye la entrega, mantenimiento y la mejora del software. (ionos, 2021)

* 1. **TECNOLOGÍA MÓVIL**

A medida que va pasando el tiempo la tecnología va evolucionando cada vez más y los dispositivos móviles son parte de ello, ya que con el tiempo esto fue mejorando cada vez más y se puede observar en la actualidad como las aplicaciones móviles facilitan muchos trabajos, ahorrando tiempo y dinero en algunos casos.

Los dispositivos móviles cuentan con muchísimas funciones muy útiles para cada entorno. Además, no hay que olvidar mencionar que los estos cuentan con una variedad de características y herramientas que se encuentran ahora en un solo dispositivo.

En la última década, las apps móviles se han posicionado como unas de las herramientas más eficaces para las empresas, especialmente en el campo del e-commerce. Y no es para menos, pues las aplicaciones facilitan la vida de los usuarios, promueven la interconectividad y mejoran la experiencia en la adquisición de productos y servicios. (economiatic 2021)

Según un estudio realizado por la compañía Flurry Analytics, el usuario promedio pasa alrededor de 5 horas diarias en su teléfono y más de la mitad de este tiempo lo dedica a las apps. En el año 2017 los ingresos a través de apps generaron un total de 76 mil millones de dólares. (economiatic 2021)

En definitiva, el uso de las apps cambió la dinámica del mercado para siempre. Por ello, la implementación de estos sistemas en los modelos de negocio es un pilar fundamental para el crecimiento de toda startup. (economiatic 2021)

* + 1. **APLICACIÓN MÓVIL**

Una aplicación móvil, también conocida como aplicación móvil, es un tipo de aplicación que se puede ejecutar en un dispositivo móvil, como un teléfono inteligente o una tableta. Incluso aquí

Si bien las aplicaciones suelen ser programas de software pequeños con funciones limitadas, pueden brindar a los usuarios servicios y experiencias de alta calidad. (anincubator 2021)

Estas aplicaciones no dependen de sistemas de software integrados, a diferencia de las aplicaciones de escritorio. Sin embargo, cada aplicación móvil ofrece una funcionalidad única y limitada.

En la actualidad se utilizan tres tipos de aplicaciones:

* Aplicaciones nativas: son aplicaciones que se desarrollan para un determinado sistema operativo en específico, llamado Software Development Kit o SDK. Por ejemplo:
  + IOS, las aplicaciones se desarrollan con lenguaje Objective-C.
  + Windows Phone, las aplicaciones se desarrollan en .Net.
  + Android, las aplicaciones se desarrollan en lenguaje Java.
* Aplicaciones hibridas: son aplicaciones que funcionan en diferentes sistemas operativos sin la necesidad de programar en diferentes lenguajes de programación para cada sistema operativo.
* Aplicaciones web: son aplicaciones que se desarrollan con lenguajes como HTML, Javascript, CSS. Una de las principales ventajas con las aplicaciones nativas es la posibilidad de programar independiente al sistema operativo que se esté usando. Estas aplicaciones web son ejecutadas dentro del propio navegador del dispositivo
  1. **ENTORNO DE DESARROLLO FLUTTER**

“Flutter es un framework de Dart para crear aplicaciones multiplataforma con un único código. A diferencia de otros frameworks multiplataforma como por ejemplo Ionic, el código de una aplicación de Flutter se compila a código nativo, por lo que el rendimiento alcanzado es superior a aplicaciones basadas en web-views. Además, a diferencia de React Native, Flutter no utiliza componentes nativos, sino que viene con sus propios componentes, llamados widgets, por lo que la misma aplicación se verá igual en cualquier dispositivo, independientemente de su sistema operativo o la versión. Gracias a ello, el desarrollador no tiene que preocuparse por que el diseño de su aplicación se vea mal en dispositivos antiguos.” (Inlab Fib 2020) La principal ventaja de este framework es que se desarrolla únicamente una vez para todos los sistemas operativos, esto quiere decir una reducción de costes y tiempo de producción a la hora de elaborar un proyecto. Los widgets son utilizados para la construcción de las interfaces, aprovechando esto también se puede realizar widgets personalizados que son más completos.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

**Figura 4: Arquitectura de Flutter**

* + 1. **FUNCIONALIDADES DE FLUTTER**

Algunas de las razones y beneficios funcionales de utilizar Flutter son las siguientes:

* Calidad nativa: Las aplicaciones nativas se desarrollan solamente para un sistema operativo. Una característica principal es que este usa todas las ventajas de las aplicaciones nativas para una mejor calidad al final del proyecto. (speakerdeck 2021)
* Experiencia de usuario: Flutter incluye Material Design de Google y Cupertino de Apple, esto ayuda a que las interfaces desarrolladas son muy parecidas a las aplicaciones que desarrollan sus propias compañías. También, esto permite que la experiencia de usuario sea mucho más óptima. (speakerdeck 2021)
* Tiempo de carga: El tiempo de carga en Flutter destaca exitosamente ya que estos se encuentran por debajo de un segundo en cualquiera de los soportes IOS o Android. (speakerdeck 2021)
* Desarrollo ágil y rápido: Con la ayuda de hot-reload, la experiencia del desarrollador es más satisfactoria, ya que esto permite ver los cambios en el dispositivo o emulador al instante de haber actualizado el código. (speakerdeck 2021)
  1. **LENGUAJE DE PROGRAMACIÓN DART**

“El lenguaje de programación Dart fue desarrollado principalmente por Google. Dart es un estándar Ecma, la organización europea para la estandarización de sistemas informáticos y de comunicación y productos electrónicos. La programación Dart es una alternativa interesante a JavaScript en los navegadores web actuales. De acuerdo con los desarrolladores de Dart, ya no es posible subsanar las deficiencias de JavaScript mediante el desarrollo del lenguaje. (ionos, 2021)

“El lenguaje Dart de Google empezó a desarrollarse en 2010 y se presentó un año después. Como los navegadores no podían, ni pueden, trabajar con este lenguaje de forma natural, y JavaScript puede ejecutarse en todos los navegadores actuales, existe el compilador Dart2js, es decir, “Dart para JavaScript”. El lenguaje Dart se asemeja a los ya establecidos lenguajes de programación orientados a objetos, entre los que se encuentran Swift, C# o Java, que se subordinan a determinados paradigmas de programación. Las reglas para combinar caracteres definidos, es decir la sintaxis, son similares al lenguaje C. Esta semejanza facilita enormemente el aprendizaje, de manera que es posible iniciarse en él sin tener que enfrentarse a grandes problemas de lenguaje.” (ionos, 2021)

Algo que caracteriza a dart es que, a diferencia de muchos lenguajes, este hace que el proceso de desarrollo sea más cómodo y rápido para los desarrolladores. Por eso, dart viene con su propio gestor de paquetes, también contiene varios compiladores. Además, su máquina virtual y la compilación Just-in-Time permiten que los cambios realizados al código se vean reflejado inmediatamente.

A continuación, algunas características del lenguaje dart:

* Es de código abierto, esto permite el acceso gratuito para cualquier persona que desea programar en este lenguaje.
* La sintaxis de este lenguaje es muy fácil de aprender.
* Funciona para todos los navegadores móviles y de escritorio actuales.
* Dart está totalmente orientado a objetos.
* Soporte de muchas herramientas de desarrollo, como intelliJ IDEA y Visual Studio

**DIAGRAMAS DE CASOS DE USO**

Este diagrama representa la funcionalidad del sistema. partir de la perspectiva de los stakeholders sobre el funcionamiento del sistema. Está diseñado para mostrar cómo las funciones interactúan con los usuarios internos y externos. Este es generalmente el diagrama con el que se inicia el análisis. Los componentes de este diagrama se muestran en la figura 2. Los casos de uso representan las funcionalidades del sistema actores que cumplen con las funciones de los usuarios y los límites del sistema que representan el alcance del caso de uso.

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza baja

*Figura 2: Elementos de Diagrama de caso de uso*

**DIAGRAMA DE SECUENCIA**

Además del orden en que suceden, analiza la secuencia de mensajes y eventos que se envían y reciben objetos. Representan las interacciones en situaciones específicas La figura 3 muestra los elementos que componen este diagrama; al igual que en el diagrama anterior, los actores representan los roles de usuario; los objetos representan todos los objetos que forman parte de la interacción; la línea de vida del objeto muestra la participación individual del objeto en la interacción; y el periodo de actividad del objeto muestra el tiempo de actividad del objeto. (Atlassian, s.f.)

Imagen en blanco y negro

Descripción generada automáticamente con confianza media

*Figura 3: Elementos que componen el diagrama de secuencia*

**DIAGRAMA DE CLASE**

Un diagrama de clases es una representación visual de los objetos de clases en un sistema de modelo, clasificados por tipos de clases. Cada tipo de clase es representada como un rectángulo con tres compartimientos para el nombre de la clase, los atributos, y las operaciones. (Atlassian, s.f.)

Texto

Descripción generada automáticamente con confianza baja

*Figura 4: Elementos de diagrama de clases*

**MODELO DE ENTIDAD RELACIÓN**

Este modelo representa la estructura lógica de una base de datos, que incluye las relaciones y limitaciones que determinan cómo se almacenan y acceden a los datos.   
Los elementos que conforman este diagrama se muestran en la Figura 5. Las entidades que guardan el nombre y la cantidad, los atributos que componen y el tipo de variable del atributo. (Atlassian, s.f.)

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

*Figura 5: Elementos del modelo entidad relación*

**MODELOS DE DISEÑO**

Es una representación conceptual o abstracta de un sistema, producto o proceso que está siendo diseñado. Antes de que se escriba el código real, se desarrolla un modelo de diseño en el diseño de software, que describe la estructura y el comportamiento del sistema que se está desarrollando. (IBM, 2017)

**DIAGRAME DE ACTIVIDADES**

El diagrama de actividades utiliza símbolos gráficos para representar acciones, decisiones, bifurcaciones y otros elementos que describen el comportamiento de un sistema en un diagrama de actividades. Estos diagramas son útiles para visualizar y comprender el flujo de trabajo de un proceso o sistema, ya que identifican las actividades que se realizan en cada etapa y las condiciones que dictan cómo se debe proceder. Los componentes de este diagrama se muestran en la Figura 6.

Forma, Polígono

Descripción generada automáticamente

*Figura 6: Elementos que compone el Diagrama de actividades*

* 1. **METODOLOGÍAS DE DESARROLLO**
     1. **BUSNESS LOGIC COMPONENT (Bloc)**

[Bloc](https://github.com/felangel/bloc/blob/master/packages/bloc/README.md) (Business Logic Component) es un componente que sirve de intermediario entre la vista y nuestro modelo de datos. Es similar al concepto de ViewModel (de la [arquitectura MVVM),](https://www.paradigmadigital.com/dev/arquitectura-mvvm-swiftui-combine/) ya que nos facilita gestionar de una forma ordenada los cambios de estado que se producen en el modelo de datos de nuestra aplicación para que, de forma reactiva, la interfaz refleje los cambios necesarios. Se basa en el patrón observer, siguiendo la mecánica de escuchar eventos, y transformarlos en nuevos estados. En la figura 7 se observa el flujo del patrón en que se diferencian tres elementos relevantes, UI ser a la vista, Bloc es el componente de lógica de negocios y finalmente los datos.

Diagrama

Descripción generada automáticamente con confianza media

Figura 7: patron Bloc (*Bloc | Dart Package*, n.d.)

* + 1. **PATRÓN POR CAPAS**

La arquitectura en capas es un modelo de diseño de software que se basa en dividir las diversas funcionalidades del sistema en capas o niveles, cada uno de los cuales se encarga de un conjunto específico de tareas y se comunica con los niveles superiores a través de interfaces claras.

La capa más utilizada es el MVC (Modelo Vista Controlador)

“MVC (Modelo-Vista-Controlador) es un patrón en el diseño de software comúnmente utilizado para implementar interfaces de usuario, datos y lógica de control. Enfatiza una separación entre la lógica de negocios y su visualización. Esta "separación de preocupaciones" proporciona una mejor división del trabajo y una mejora de mantenimiento. Algunos otros patrones de diseño se basan en MVC, como MVVM (Modelo-Vista-modelo de vista), MVP (Modelo-Vista-Presentador) y MVW (Modelo-Vista-Whatever). (Mdn, 2023)”

* 1. **(CONCEPTOS Y DEFINICIONES UTILIZADAS DEL PROYECTO)**
  2. **MARCO INSTITUCIONAL**

1. **MARCO PRÁCTICO**
   1. **ANÁLISIS PRELIMINAR**

El análisis preliminar tiene como objetivo evaluar la situación actual del control y seguimiento de inventarios para equipos electrónicos en el Departamento de Centro de Sistema de la Universidad Católica Boliviana "San Pablo". Se buscan los principales problemas y mejoras en el proceso de gestión de inventarios recopilar información y evaluar las operaciones actuales. Los registros de inventario, los documentos internos y los procedimientos operativos actuales se revisaron para recopilar información. Además, se realizaron entrevistas con el personal del departamento responsable del manejo de inventarios para conocer sus experiencias y percepciones sobre el proceso actual.

* 1. **DETERMINACIÓN DE REQUERIMIENTOS**

En la presente sección se presentarán los requerimientos funcionales y los requerimientos no funcionales del proyecto.

* + 1. **REQUERIMIENTOS FUNCIONALES**

En la tabla 1 se muestran los requerimientos funcionales del proyecto

|  |  |  |  |
| --- | --- | --- | --- |
| **Número** | **Requerimiento** | **Descripción** | **Prioridad** |
| **RF-1** | Inicio  de sesión | El administrador debe ingresar al sistema con Google | Alta |
| **RF-2** | Registro  de equipos | El sistema permitirá  El registro de equipos electrónicos, incluyendo información detallada como modelo, marca, número de serie, etc.  Generar un código de identificación único para cada equipo registrado. | Medio |
| **RF-3** | Asignación de equipo | El sistema de App. móvil permitirá:  La asignación de equipos a miembros del personal administrativo y académico.  Registrar información sobre el usuario asignado y la fecha de asignación. | Medio |
| **RF-4** | Seguimiento de equipos | El sistema de App, móvil permitirá:  Realizar un seguimiento del estado y ubicación de cada equipo registrado.  Generar alertas cuando un equipo requiera mantenimiento o presente fallas. | Alto |
| **RF-5** | Control de mantenimientos | El sistema de App. móvil permitirá:  Programar mantenimientos preventivos para los equipos.  Registrar el historial de mantenimientos realizados a cada equipo. | Medio |
| **RF-6** | Inventario de equipos | El sistema de aplicación móvil permitirá:  Mantener un inventario actualizado de todos los equipos electrónicos de la universidad.  Generar informes detallados del inventario, incluyendo filtros y búsquedas. | Medio |
| **RF-7** | Control de repuestos: | El sistema de aplicación móvil permitirá:  Llevar un control de los repuestos disponibles para los equipos. | Alto |
| **RF-8** | Cerrar sesión | El sistema de aplicación móvil permitirá cerrar las sesiones en un determinado tiempo | Medio |
| **RF-9** |  |  |  |
| **RF-10** |  |  |  |
| **RF-11** |  |  |  |
| **RF-12** |  |  |  |
| **RF-13** |  |  |  |

Tabla 1: Requerimientos funcionales

* + 1. **REQUERIMIENTOS NO FUNCIONALES**

La tabla 2 muestra los requerimientos no funcionales del sistema; el orden de las columnas es idéntico al de la tabla 1, que muestra los requerimientos funcionales, pero no se les otorgó prioridad.

|  |  |  |
| --- | --- | --- |
| **Número** | **Requerimiento** | **Descripción** |
| **RNF-1** | Disponibilidad de la aplicación | Garantizar una alta disponibilidad de la aplicación y los servicios asociados.  Implementar medidas de respaldo y recuperación de datos. |
| **RNF-2** | Seguridad | Encriptar datos sensibles como información de usuarios y equipos durante la transmisión y almacenamiento. |
| **RNF-3** | Compatibilidad | Asegurar la compatibilidad de la aplicación con las últimas versiones de los sistemas operativos móviles Android. |
| **RNF-4** | Mantenibilidad | Implementar una arquitectura modular y bien documentada para facilitar el mantenimiento y las futuras actualizaciones. |

Tabla 2: Requerimiento no funcionales

* 1. **ANÁLISIS DEL PROYECTO**
     1. **MAQUETADO (Wireframing)**
     2. **EJECUCIÓN DE LAS ETAPAS CORRESPONDIENTES A LA METODOLOGÍA**
     3. **HISTORIAS DE USUARIO**
     4. **MODELOS UML**
        1. **CASOS DE USO**

Los diagramas de casos de uso permiten modelas la funcionalidad del sistema para describir las acciones provistos dentro del sistema para los actores. Un caso de uso describe el procedimiento en detalle, lo cual ayudara a tener una mejor comprensión de las tareas que pueden realizar los actores. (Jacobson, Rumbaugh y Booch, 2000). Para la elaboración de los siguientes diagramas de casos de uso se tomaron en cuenta los siguientes actores: administrador aplicación, administrador proyecto, jefe de obra, encargado de obra

Diagrama

Descripción generada automáticamente

**Figura 7: Actores del sistema móvil**

En la Figura 7 se puede observar los actores que serán parte del sistema móvil, en total serán cuatro los cuales serán el administrador de la aplicación quien se encargará de actualizar los precios de material de la base de datos, el administrador de proyectos, quien es encargado de crear proyectos, crear usuarios, jefe de obra y encargado de obra, crear presupuesto. También est00E1 el jefe de obra quien es el encargado de iniciar las etapas del proyecto y darle un respectivo seguimiento. El encargado de obra quien es el encargado de hacer seguimiento a cada etapa de proyecto.

* + - 1. **INICIAR SESIÓN**

El diagrama correspondiente al caso de uso de iniciar sesión es el siguiente

Diagrama

Descripción generada automáticamente

Figura 8: Diagrama de caso de uso 1 -Iniciar sesión

En la figura 8 se puede observar que el sistema de aplicación móvil deberá verificar las credenciales de autenticación del usuario. En caso de que estas sean correctas el sistema de aplicación móvil deberá genera la autenticación con Google.

Tabla 2: Descripción del caso de uso Iniciar sesión

|  |  |  |
| --- | --- | --- |
| **Caso de uso** | **Iniciar sesión** | |
| **Actores** | Encargado del centro de sistemas | |
| **Descripción** | Permite al usuario ingresar a la aplicación móvil | |
| **Precondiciones** | El usuario deberá contar con correo electrónico para poder ingresar a la aplicación | |
| **FLUJO NORMAL ENVENTOS** | | |
| **ACCIÓN DE ACTORES** | | **RESPUESTA DEL SISTEMA** |
| 2. El usuario ingresa a la opción de “Continuar con Google”  4. El usuario debe elegir su correo de Gmail para poder iniciar sesión  6.El usuario presiona el botón de “continuar” | | 1.El sistema muestra el inicio de sesión con Google  3.El sistema muestra el/los correos que tiene el usuario  5.El sistema verifica las credenciales de autenticación  7.Ingresa a la aplicación móvil |
| **Flujo alternativo:**  Las credenciales son incorrectas el sistema muestra un mensaje de error | | |

* + - 1. **REGITRAR PERSONAL**

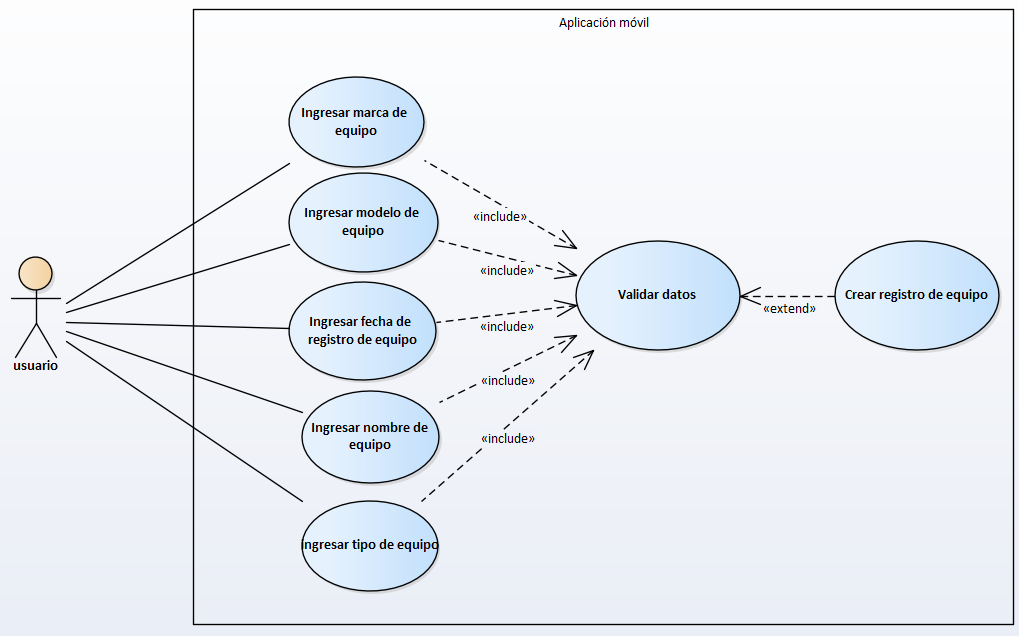


Figura 9: Diagrama de caso de uso 2 Registrar personal

En la figura 9 se muestra el caso de uso correspondiente al registro personal. Para ello el sistema de aplicación móvil deberá validar los datos introducidos en el formulario de registro y si estos son correctos deberá crear un nuevo personal lo que requiere guardar los datos del usuario en la base de datos.

|  |  |  |
| --- | --- | --- |
| **Caso de uso** | **Registrar Personal** | |
| **Actores** | Encargado de centro de sistemas | |
| **Descripción** | Permite al administrador registrar un nuevo personal | |
| **Precondiciones** |  | |
| **FLUJO NORMAL ENVENTOS** | | |
| **ACCIÓN DE ACTORES** | | **RESPUESTA DEL SISTEMA** |
| 1.Ingresar al menú de registrar  3.Ingresar datos del usuario  5.El usuario presiona el botón de guardar | | 2.vizualizar formulario de registro  4.validar datos del ingresados  6.Registro guardado |
| **Flujo alternativo:**  Los datos de formulario son incorrectos el sistema no guardara el registro y mostrara un mensaje de error | | |

* + - 1. **REGISTRAR EQUIPO**

Diagrama

Descripción generada automáticamente

**Figura Diagrama de caso de uso 3:** Registrar equipo

|  |  |  |
| --- | --- | --- |
| **Caso de uso** | **Registrar Personal** | |
| **Actores** | Administrador de centro de sistemas | |
| **Descripción** | Permite al administrador registrar un nuevo equipo | |
| **Precondiciones** | El nuevo equipo debe tener código de barra o código serial | |
| **FLUJO NORMAL ENVENTOS** | | |
| **ACCIÓN DE ACTORES** | | **RESPUESTA DEL SISTEMA** |
| 1.Ingresar al menú de registrar equipo  3.Ingresar datos del nuevo equipo  5.El administrador presiona el botón de guardar | | 2.mostrar formulario de registro de equipo  4.Validar datos del formulario  6. Mensaje de registro de equipo guardado |
| **Flujo alternativo:**  Los datos del formulario de registro son incorrectos se mostrara un mensaje de error | | |

* + - 1. **ASIGNAR EQUIPO**

Diagrama

Descripción generada automáticamente

Figura Diagrama de caso de uso 4: Asignar equipo

|  |  |  |
| --- | --- | --- |
| **Caso de uso** | **Asignar equipo** | |
| **Actores** | Administrador de sistema | |
| **Descripción** | Permite al administrador asignar un nuevo equipo | |
| **Precondiciones** |  | |
| **FLUJO NORMAL ENVENTOS** | | |
| **ACCIÓN DE ACTORES** | | **RESPUESTA DEL SISTEMA** |
| 1.Ingresar al menú de asignar equipo  3.ingresar código de equipo para signar  4.Ingresar código de nuevo usuario para asignar  5.Ingresar fecha  7.El administrador presiona el botón de guardar | | 2.Mostrar formulario de asignar equipo  6.Validar datos del formulario de asignar  8.Mensaje de asignar nuevo equipo al usuario registrado |
| **Flujo alternativo:**  Si el código de equipo es duplicado entonces se mostrara un mensaje de error | | |

* + - 1. **MANTENIMIENTO DE EQUIPO**

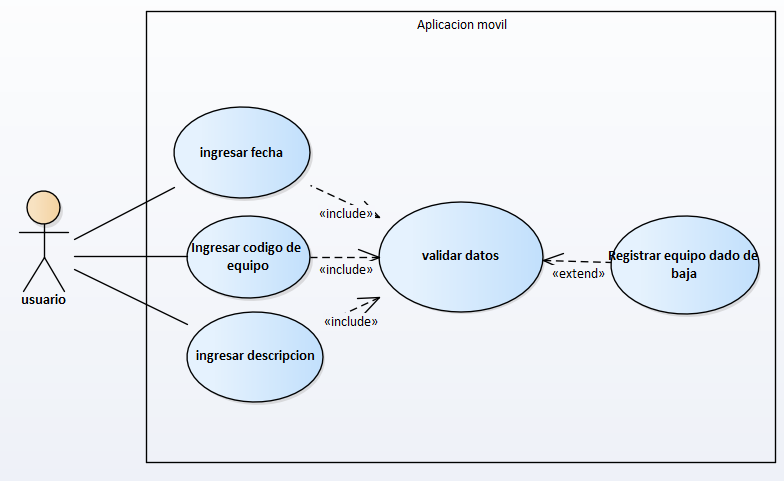
Diagrama

Descripción generada automáticamente

**Figura Diagrama de caso de uso 5: Mantenimiento de equipo**

|  |  |  |
| --- | --- | --- |
| **Caso de uso** | **Mantenimiento de equipo** | |
| **Actores** | Administrado del centro de sistemas | |
| **Descripción** | El administrador podrá registrar el mantenimiento de equipo | |
| **Precondiciones** | El usuario debe enviar un email al administrador para solicitar el mantenimiento | |
| **FLUJO NORMAL ENVENTOS** | | |
| **ACCIÓN DE ACTORES** | | **RESPUESTA DEL SISTEMA** |
| 1.Ingresar al menú de mantenimiento  3.Ingresar código de equipo  4.Ingresar nombre de usuario  5.Ingresar descripción de equipo  6.Ingresar fecha de mantenimientos  8.El administrado presiona el botón de registrar manteamiento | | 2.Mostrar formulario de registro de mantenimiento  7.Validar datos de registro de mantenimiento  9.Visualizar mensaje de registro guardado |
| **Flujo alternativo:** | | |

* + - 1. **DAR DE BAJA UN EQUIPO**



**Figura Diagrama de caso de uso 6: Dar de baja un equipo**

|  |  |  |
| --- | --- | --- |
| **Caso de uso** | **Dar de baja un equipo** | |
| **Actores** | Administrado de centro de sistemas | |
| **Descripción** | El administrado podrá realizar dar de baja el equipo | |
| **Precondiciones** | El equipo debe tener código serial | |
| **FLUJO NORMAL ENVENTOS** | | |
| **ACCIÓN DE ACTORES** | | **RESPUESTA DEL SISTEMA** |
| 1.Ingresar al menú de mantenimientos  2.ingresar al submenú de dar de baja equipo  4.Ingresar código de quipo  5.Ingresar motivo de baja  6.Ingresar fecha | | 3.Visualizar formular de baja de equipo  7Muestra mensaje de equipo dado de baja |
| **Flujo alternativo:** | | |

* + - 1. **SEGUMIENTO DE EQUIPO**
      2. **REPORTES**

Diagrama

Descripción generada automáticamente

**Figura Diagrama de caso de uso: Generar reporte**

|  |  |  |
| --- | --- | --- |
| **Caso de uso** | **Reporte** | |
| **Actores** | Administrado de centro de sistemas | |
| **Descripción** | El administrado podrá visualizar los reportes | |
| **Precondiciones** |  | |
| **FLUJO NORMAL ENVENTOS** | | |
| **ACCIÓN DE ACTORES** | | **RESPUESTA DEL SISTEMA** |
| 1.Ingresar al menú de reporte  2.Selecionar un filtro de reporte  3.Seleccionar el tipo de reporte | | 4.Visualizar reporte |
| **Flujo alternativo:** | | |

* + - 1. **REPORTE DE EQUIPOS DADOS DE BAJA**
      2. **DIAGRAMA DE ACTIVIDADES**
         1. **Inicio de Sesión**

El diagrama de actividades correspondiente al caso de uso de inicio de sesión es:

Diagrama

Descripción generada automáticamente

Figura Diagrama de Actividades de inicio de sesión

* + - * 1. **Diagrama de Registro de usuario**

El siguiente diagrama de actividades corresponde al registro del Docente en la aplicación:

Diagrama

Descripción generada automáticamente

Figura Diagrama de actividades de registro de usuario

* + - * 1. **Diagrama de Registro de Equipos**

El siguiente diagrama de actividades representa el registro de dispositivos electrónicos en la aplicación móvil:

Diagrama

Descripción generada automáticamente

* + - * 1. **Diagrama de Mantenimiento**

El siguiente diagrama de actividades representa al mantenimiento de dispositivos electrónicos en la aplicación:

Diagrama

Descripción generada automáticamente

Figura: Diagrama de actividades: de mantenimiento de equipos.

* + - * 1. **Diagrama de seguimiento.**
        2. **Diagrama de Asignación de Equipo.**

El siguiente diagrama de actividades representa al asignar un dispositivo electrónico de la aplicación:

Diagrama

Descripción generada automáticamente

Figura

* + - * 1. **Diagrama de reportes**

El siguiente diagrama de actividades representa al reporte de los dispositivos electrónicos de la aplicación:

Diagrama

Descripción generada automáticamente

* + 1. **DIAGRAMA DE COMPONENTES**
    2. **INTERFACES**
  1. **DISEÑO DEL PROYECTO**
     1. **MODELO DE ARQUITECTURA C4**
     2. **MAQUETADO**
     3. **EJECUCIÓN PRELIMINAR DEL PROTOTIPO**
  2. **BASE DE DATOS**
  3. **VALIDACIÓN Y PRUEBAS DEL SISTEMA**
  4. **ANÁLISIS DE COSTOS**
  5. **DESARROLLO DEL PROTOTIPO**

1. **CONCLUCIONES Y RECOMENDACIONES**

**CONCLUSIONES**

* 1. **RECOMENDACIONES**

1. **BIBLIOGRAFIA**
2. **ANEXOS**